Breaking license

Artem Bachevsky

HighLoad++ Armenia

Co-organizer

Yandex

whoami

- Software developer -> AppSec Expert
- Licensed software user
- Cybersecurity researcher
- @frydaykg

What we’ll talk about

- What, from what, and how we are protecting?
- And how do they break us?
- And what can we resist in response?

From what?

- Protection against unauthorized use of programs is a system of measures aimed at countering the illegal use of software. When protecting, organizational, legal, software and software-hardware means can be used.(c) Wikipedia

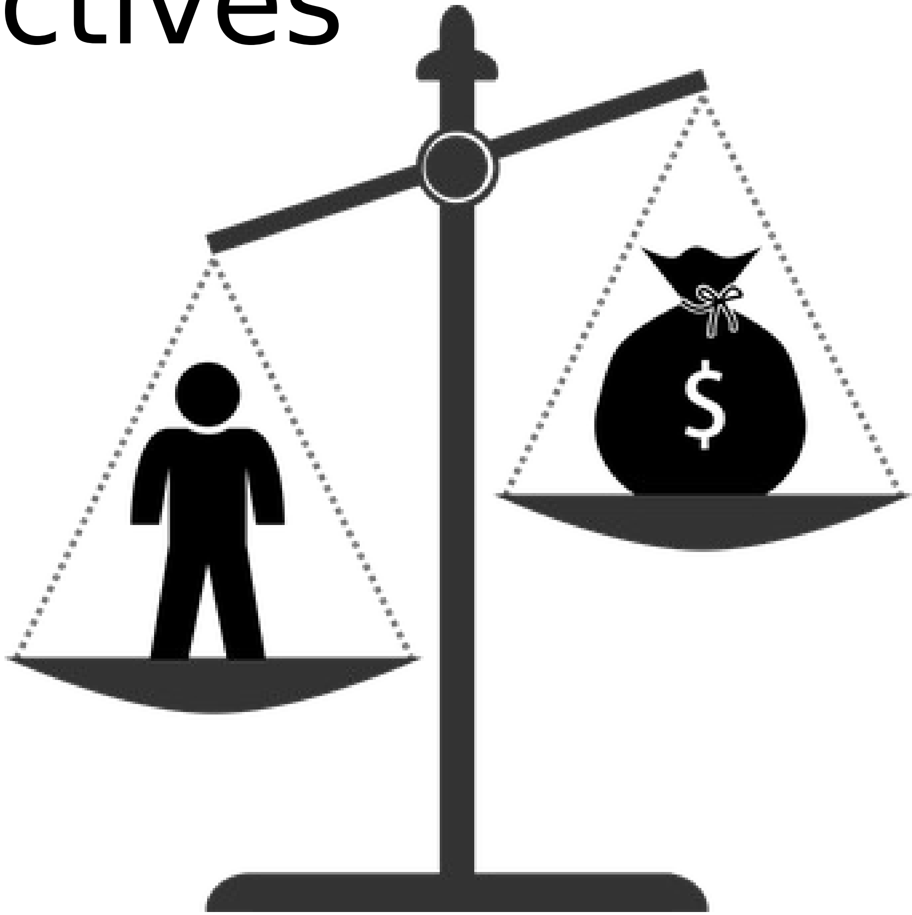
What?

- The ability to use the software
- The ability to use the software in agreed time intervals
- Paid functionality
- Our specific restrictions and limits

Basic principles and objectives

Choose a ratio of protection measures such as:

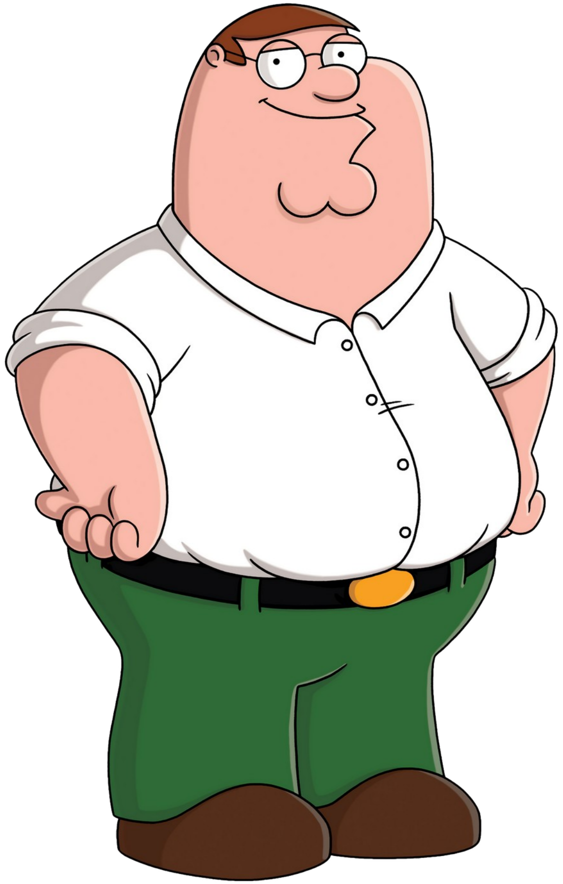
- User UX doesn't suffer much
- It is expensive to break the defense
- And user would be willing to pay...



Software activation

By object of applicability:

- Thick client
- Thin client(web)



Software activation

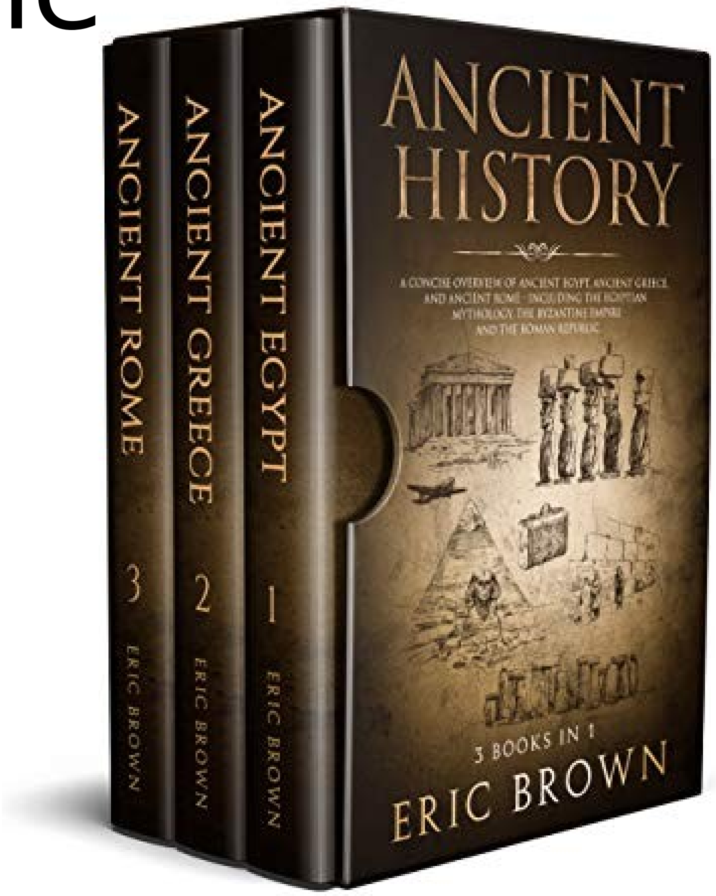
By approach:

- Offline
- Online
- Local activation server

Activation approach: offline

Attacks

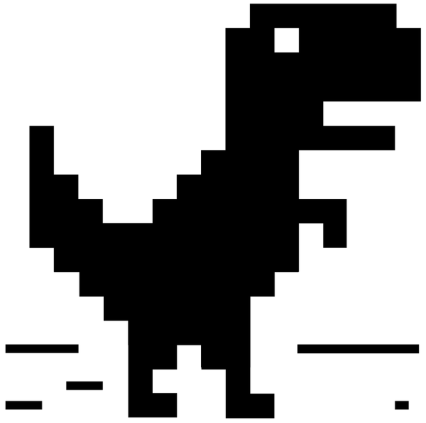
- Via serial key distribution


ANCIENT HISTORY
ANCIENT ROME 3 ERIC BROWN
ANCIENT GREECE 2 ERIC BROWN
ANCIENT EGYPT 1 ERIC BROWN
3 BOOKS IN 1
ERIC BROWN

Activation approach: offline

But if must use offline activation then:

- Use unique installers
- Develop telemetry system
- Upgrade your EULA and organizational measures



You are offline

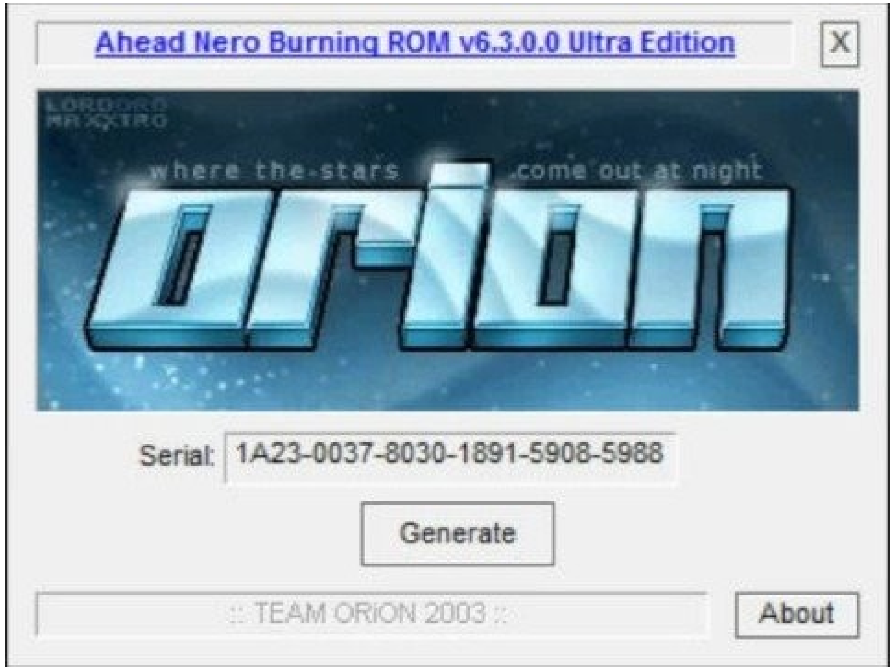
Activation approach: online

1. Fingerprint is generated
2. Fingerprint gets to vendor
3. Vendor returns activation code
4. Code is entered into the program

Activation approach: online

Attacks

- Keygens
- Patching
- Attacks on activation server
- Environment emulation


Ahead Nero Burning ROM v6.3.0.0 Ultra Edition
X
where the stars come out at night
orion
Generate
:: TEAM ORiON 2003 ::
About

Serial: 1A23-0037-8030-1891-5908-5988

Case study: attack on activation server

- Online activation
- Generating a signature hash of hardware => license
- Check for hardware spoofing

What could possibly go wrong?

Case study: attack on activation server

Hacked by PhyRo

© file in the root directory of server

Case study: attack on activation server

A chain is as strong as the weakest link ©

Keep in mind:

- Infrastructure security
- Third-party dependencies

Software activation

By uniqueness of an object:

- Unique installer
- Hardware
- OS user profile
- Application account

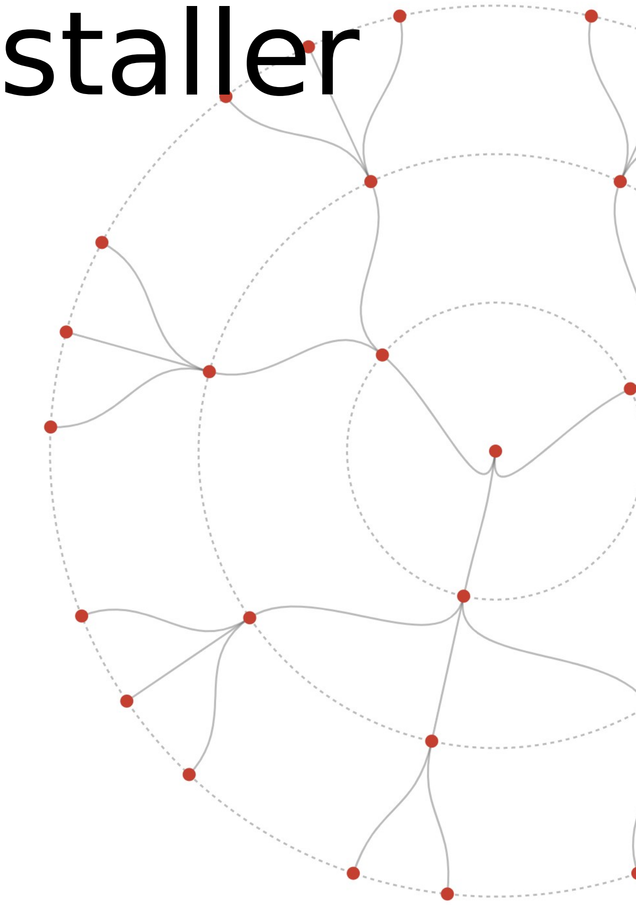
Software activation: unique installer

Ideal best case:

- Online activation
- Periodic online check

Otherwise it will be a failure.

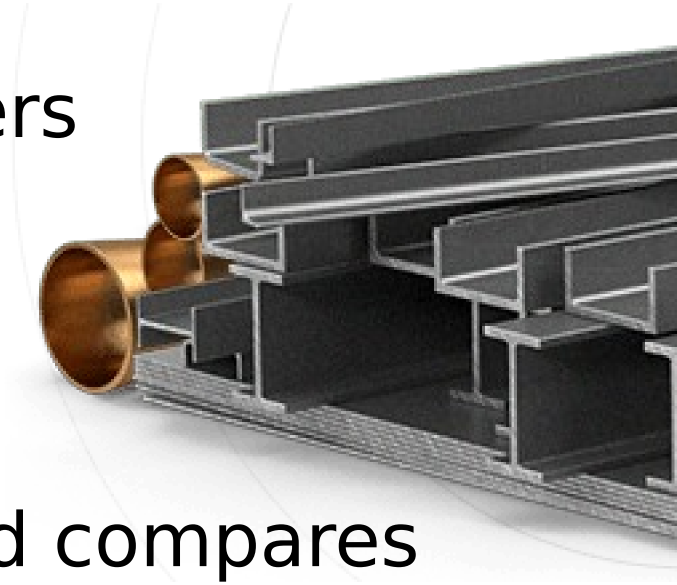
But now it is possible to track software spreading!



Software activation: hardware

Principle:

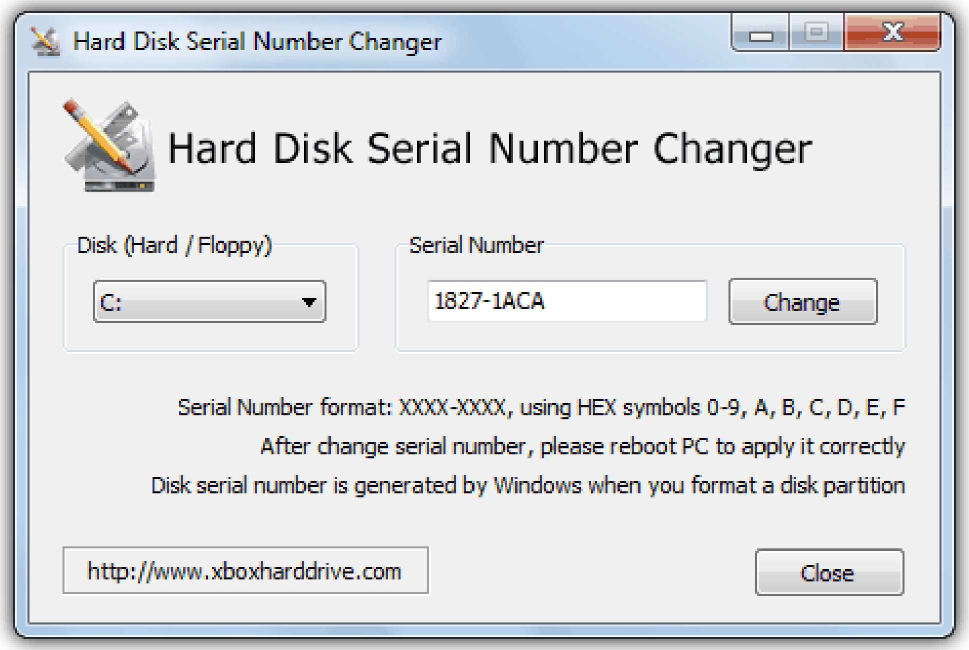
1. Collect a set of unique hardware parameters
2. Hash them
3. Vendor signs a hash
4. Signature is a license
5. Software periodically generates a hash and compares it to the signature



Case study: hardware spoofing

- Thick client
- Hardware binding
- Semi-online activation

Looks good, doesn’t it?



Hard Disk Serial Number Changer

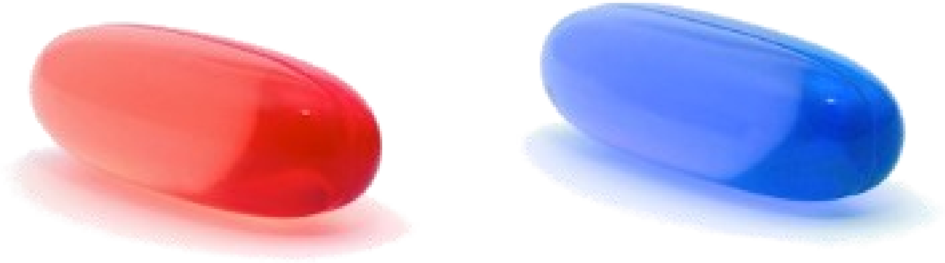
Disk (Hard / Floppy): C:

Serial Number: 1827-1ACA

Serial Number format: XXXX-XXXX, using HEX symbols 0-9, A, B, C, D, E, F
After change serial number, please reboot PC to apply it correctly
Disk serial number is generated by Windows when you format a disk partition

Case study: hardware spoofing

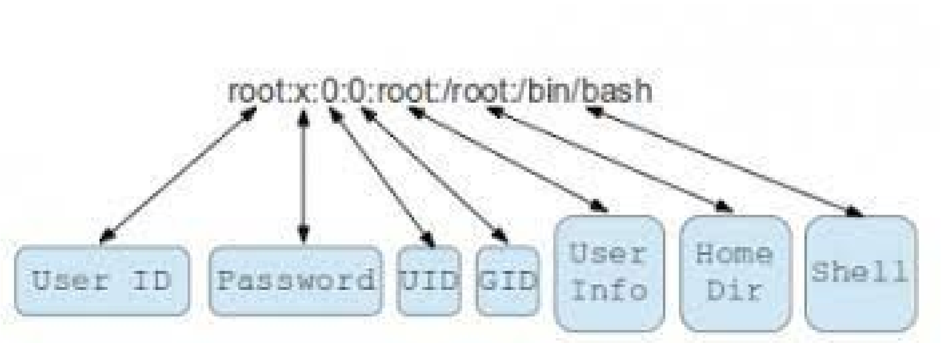
- Weak hardware metadata
  - Does a strong metadata exist at all?
- Ability to run in virtual environment
  - Red Pill



Software activation: OS user profile

We allow only one user per license to use the software, but also on multiple devices.

Binding objects: OS user, its metadata



Software activation: OS user profile

In reality, in addition it takes into account:

- Number of requested activations on various devices during the period
- Similarity of usernames

Attacks

- Runtime emulation (OS username, transfer of license files)
- Fraud with the number of activations per license

Case study: license transfer

- A popular AppSec tool
- License data stored in registry or settings file

How to break it?

Case study: license transfer

- Track files and registry changes
  - Process monitor/strace
- Calculate diff before and after activation
- Find out the exact metadata for binding
  - Eyes
  - Decompilers
- Make a patch for the registry, OS, file system
- PROFIT!!1

Software activation: application account

- Applicable to thin clients
- Almost always solutions require access to the Internet
- Activation = the fact of signing-in the service with a specific login

Software activation: application account

My ideal licensing system:

- We work in a thin client (web)
- Licenses are purchased per user

Where should I look as an attacker?

Software activation: application account

Software activation: application account

- Vulnerabilities
  - Checking permissions on the frontend
  - IDOR
  - Broken Access Control
  - …
- Works on one account for many users
- Works for many users through a single proxy server

Software activation: application account

- AppSec practices
- Security audit
- Focus on business logic vulnerabilities in licensing issues
- Behavioral analysis
- Activity analysis

Forward to the past

Forward to the past

- System software malfunction
    - Spoofing system time for a process
    - RunAsDate utility
- Checking with Internet time sources
- Protection by metadata
    - Issue date stamps on the license itself
    - Flags with timestamps on filesystem
    - Blocking the license in case of violation of the rules

You don't bring a knife to a gunfight

If there is protection, then it can always be bypassed.

Tools:

- IDA Pro
- Ghydra
- Hopper
- Radare2
- ApkTool

Case study: binary patching

Coolest action camera vendor sells video stabilization functionality

Case study: binary patching

1. Object analysis
2. Decompilation
3. Patching

Case study: binary patching

Tools are your friends

- file
- dotPeek
- .net Reflector
- dnSpy

You don't bring a knife to a gunfight: protection

- Obfuscation
- Binary signing
- Executable packers
- Polymorphic software

Cons

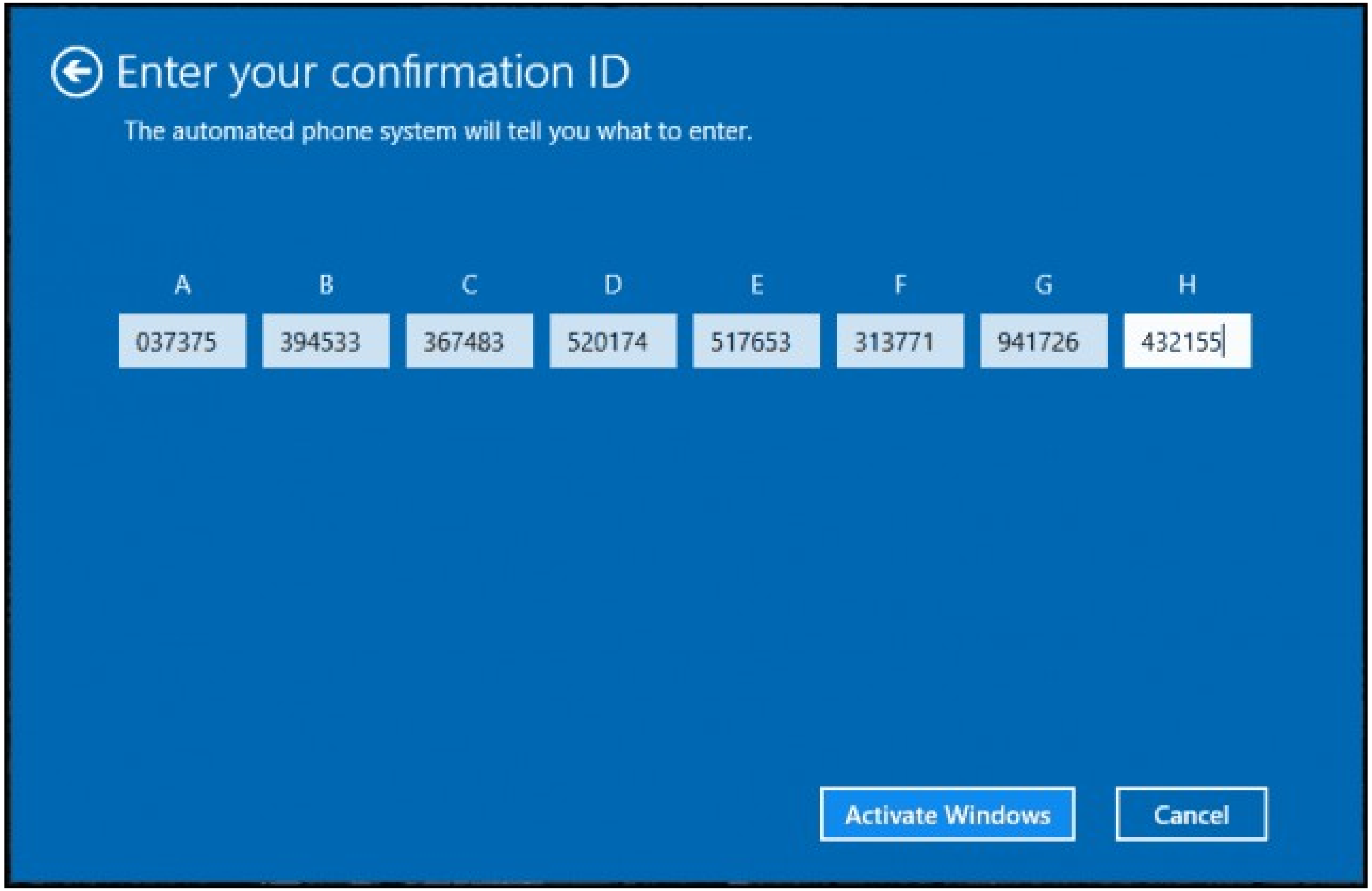
- Not a panacea
- There is a chance of making the quality worse

Case study : it's not always about technology

- Windows of higher versions
- You can live without activation but…

Case study : it's not always about technology

Case study : it's not always about technology



Enter your confirmation ID

The automated phone system will tell you what to enter.

A	B	C	D	E	F	G	H
037375	394533	367483	520174	517653	313771	941726	432155

Case study : it's not always about technology

- Think about all possible process branches
- Low-hanging fruits will be picked first
- Or even dig up potatoes

And how to live further?

- Denial, Anger, …, Acceptance
- Know your user segment
- Choose a protection model depending on specific risks

And how to live further with on-premise?

- Binding to hardware
- Semi-online activation
- New version means new license
- Code obfuscation and executable packers

And how to live further with online?

- Thin clients solve all problems
- But be aware of AppSec and business logic errors
- It's time for everyone to be browser based IMHO
- But if it's not applicable to you, then:
  - Keep track of the number of concurrently used instances
  - Analyze their behavior
  - Take organizational measures

Artem @frydaykg Bachevsky

Leave your feedback!
You can rate the talk and give a feedback on what you've liked or what could be improved

++ High Load Armenia

Co-organizer

Yandex